The VINEYARD Approach: Versatile, Integrated, Accelerator-Based, Heterogeneous Data Centres.

The VINEYARD approach: Versatile, Integrated, Accelerator-based, Heterogeneous Data Centres

Christoforos Kachris[1], Dimitrios Soudris[1], Georgi Gaydadjiev[2], Huy-Nam Nguyen[3], Dimitrios S. Nikolopoulos[4], Angelos Bilas[5], Neil Morgan[6], Christos Strydis[7], Christos Tsalidis[8], John Balafas[9], Ricardo Jimenez-Peris[10], and Alexandre Almeida[11]

[1] Institute of Computer and Communications Systems (ICCS), GR
[2] Maxeler Technologies, UK
[3] Bull Systems, FR
[4] Queens University of Belfast (QUB), UK
[5] Foundation for Research and Technology (FORTH), GR
[6] The Hartree Centre, UK
[7] Neurasmus BV, NL
[8] Neurocom Luxembourg, LU
[9] ATHEX, GR
[10] LeanXcale, ES
[11] Globaz, PT

Abstract. Emerging web applications like cloud computing, Big Data and social networks have created the need for powerful centres hosting hundreds of thousands of servers. Currently, the data centres are based on general purpose processors that provide high flexibility buts lack the energy efficiency of customized accelerators. VINEYARD aims to develop an integrated platform for energy-efficient data centres based on new servers with novel, coarse-grain and fine-grain, programmable hardware accelerators. It will, also, build a high-level programming framework for allowing end-users to seamlessly utilize these accelerators in heterogeneous computing systems by employing typical data-centre programming frameworks (e.g. MapReduce, Storm, Spark, etc.). This programming framework will, further, allow the hardware accelerators to be swapped in and out of the heterogeneous infrastructure so as to offer high flexibility and energy efficiency. VINEYARD will foster the expansion of the soft-IP core industry, currently limited in the embedded systems, to the data-centre market. VINEYARD plans to demonstrate the advantages of its approach in three real use-cases a) a bio-informatics application for high-accuracy brain modeling, b) two critical financial applications, and c) a big-data analysis application.

Keywords: hardware accelerators, data centre, heterogeneous, big data

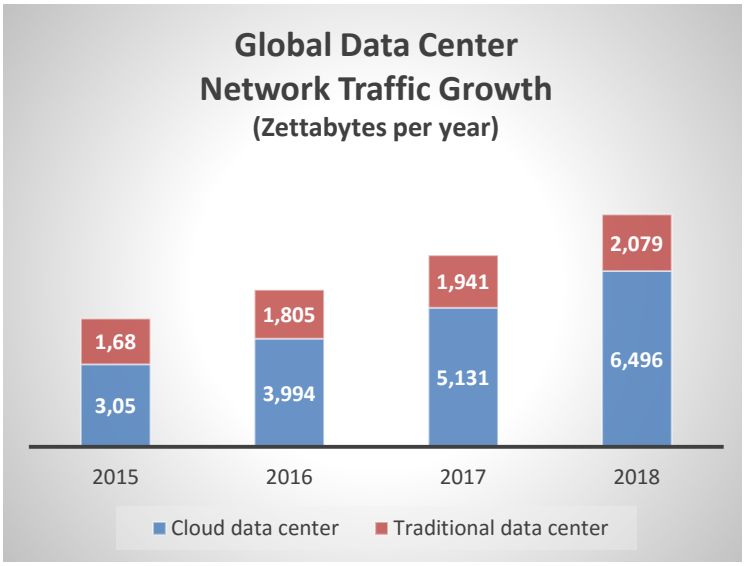

Fig. 1. Network-traffic projections for traditional and cloud-based data centres. By 2018, more than three quarters (78%) of workloads will be processed by cloud data centers; 22% will be processed by traditional data centers. [Source: Cisco Global Cloud Index]

1 Introduction

Cloud computing, Big Data and social networks are some of the emerging web applications responsible for the significant increases in data-center workloads during the last years. In 2015, the total network traffic of the data centres was around 4.7 Exabytes and it is estimated that by the end of 2018 it will cross the 8.5-Exabyte mark, following a cumulative annual-growth rate (CAGR) of 23% [1] (Figure 1). In response to this scaling in network traffic, data-centre operators have resorted to utilizing more powerful servers. Relying on Moore's law for the extra edge, CPU technologies have scaled in recent years through packing an increasing number of transistors on chip, leading to higher-performance ratings. However, on-chip clock frequencies were unable to follow this upward trend due to strict power-budget constraints. Thus, a few years ago a paradigm shift to multicore processors was adopted as an alternative solution for overcoming the problem. With multicore processors one could increase server performance without increasing their clock frequency. Unfortunately, this solution was soon found to scale poorly in the longer term, as well. The performance gains achieved by adding more cores inside a CPU come at the cost of various, rapidly scaling complexities: inter-core communication, memory coherency and, most importantly, power consumption [2].

In the early technology nodes, advancing from one node to the next allowed for a near doubling of the transistor frequency, and, by reducing the voltage,

power density remained nearly constant. With the end of Dennard scaling, advancing from one node to the next still leads to an increase in transistor density, but their maximum frequency remains roughly the same and the voltage does not decrease accordingly. As a result, the power density increases now with every new technology node. The biggest challenge, therefore, now consists of reducing power consumption and energy dissipation per mm^2 of chip area. The failure of Dennard scaling, to which the shift to multicore chips is partially a response, may soon limit multicore scaling just as single-core scaling has been curtailed. This issue has been identified in the literature as the dark-silicon era in which some of the areas in a chip are kept powered down in order to comply with thermal constraints [3].

A solution that can be used to overcome this problem is the use of application-specific accelerators. Specialized multicore processors with application-specific acceleration modules can leverage the underutilized die area to overcome the initial power barrier, delivering significantly higher performance for the same power envelope [4]. The main idea is to use the abundant die area by implementing application-specific accelerators and dynamically powering up only those accelerators suitable for a given workload. This approach can be applied either at fine-grain level (using accelerators inside the chip) or at coarse-grain level (using rack-based accelerators). In the latter case, the accelerators can either be located on the same board with the server processor or in a different blade/rack. The use of highly specialized units designed for specific workloads can greatly enhance server processors and can also increase significantly the performance of data centres subject to a maximum power budget.

This paper presents an overview of the VINEYARD H2020 project towards the development of an integrated platform for the efficient utilization of hardware accelerators in the data centres. VINEYARD aims to develop an integrated platform for energy-efficient data centres based on new servers with novel, coarse-grain and fine-grain, programmable hardware accelerators. It will, also, build a high-level programming framework for allowing end-users to seamlessly utilize these accelerators in heterogeneous computing systems by using typical data-centre programming frameworks (e.g. MapReduce, Storm, Spark, etc.).

2 VINEYARD objectives

Today's data centres consist of homogeneous processing systems (general-purpose processors) and process high volumes of data by consuming excessive amounts of power. Future heterogeneous data centres consisting of different kinds of accelerators (FPGAs, GPUs, etc.) will be able to provide higher performance under lower power consumption. However, to maintain in such heterogeneous systems the ease of programming of homogeneous ones, an integrated run-time scheduler and manager will be required to hide low-level details and relieve the user from the programming complexities involved (per different accelerator type). VINEYARD will aim specifically at the automatic utilization of accelerators through developing such an *integrated framework* that will control the hardware accel-

erators while the user will still be allowed to use typical parallel-programming frameworks.

VINEYARD will develop an *energy-efficient, integrated platform* for data centres that will consist of (1) energy-efficient servers based on customized hardware accelerators (novel programmable dataflow engines and FPGA-based servers), and (2) a software framework that will allow users to seamlessly utilize hardware accelerators in heterogeneous computing systems by using traditional data-centre and multi-core programming frameworks (e.g. MapReduce, Storm, Spark, etc.).

More specifically, the VINEYARD project will develop novel servers based on *programmable dataflow accelerators* that can be customized based on the data-centres application requirements. These programmable dataflow accelerators will be used not only to increase the performance of servers but also to reduce the energy consumption in data centres. Furthermore, VINEYARD will develop a *programming framework* that will hide the complexity of programming heterogeneous systems while at the same time providing the optimized performance of customized and heterogeneous architectures. The programming framework will leverage workload-specific accelerators based on the application requirements in a seamless fashion. The idea is that the user will work with familiar programming frameworks (e.g. MapReduce, Storm, Spark, etc.) while a *run-time manager* selects appropriate accelerators based on application requirements such as execution time, power consumption, fault tolerance and security (Figure 2). Finally and as part of the software framework, VINEYARD will provide the necessary *middleware* that binds together servers with accelerators. Along this task, VINEYARD will consider both *physical servers* and *virtual machines (VMs)*. The middleware shall also handle QoS (Quality-of-Service) concerns that arise with the shared use of the accelerators.

Figure 2 depicts the high-level diagram of the VINEYARD framework. Applications that are targeting heterogeneous data centres using traditional servers or micro-servers are programmed using traditional data-centre frameworks, such as MapReduce, and widely used data-management technologies, such as SQL (both OLTP and OLAP for operational databases and data warehouses), NoSQL (a key value data store), and Complex Event Processing (CEP). However, some of the tasks are common across several applications such as data sorting, key/value processing, encryption, compression, pattern matching, and so on, and are extremely computationally intensive. These tasks can be implemented in hardware as customized intellectual-property (IP) accelerators that can achieve much higher performance with lower power consumption. The implementation of the hardware accelerators can be achieved using traditional hardware-description languages (VHDL, Verilog) or other high-level (OpenCL) or domain-specific languages (i.e. OpenSPL). These hardware accelerators can be hosted in a repository that will interface with the run-time scheduler.

The main objectives of the VINEYARD project are:

– **Objective 1**: *Development of novel Programmable Dataflow Engines (DFE) for servers:* One of the main objectives of VINEYARD will be the devel-

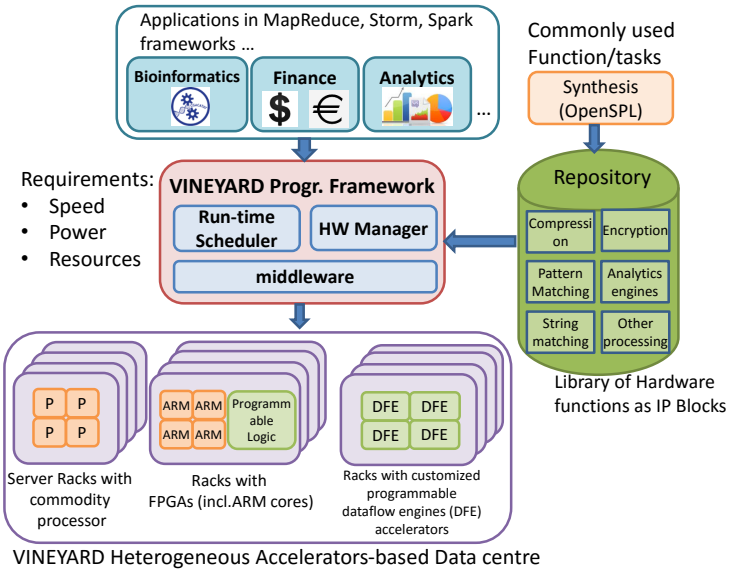

Fig. 2. High-level block diagram of the VINEYARD integrated framework. Different blocks can be seen: VINEYARD programming framework and middleware (red box), traditional servers and accelerator-based servers (purple boxes) and hardware-IP repository (green cylinder).

opment of novel programmable dataflow engines (hardware accelerators) based on coarse-grain programmable components that can be coupled to servers processor in heterogeneous data centres. The integration of the programmable hardware accelerators and traditional processors will produce integrated, high-performance and energy-efficient heterogeneous servers leading to more energy efficient data centres with higher processing power.

– **Objective 2**: *Development of novel FPGA-accelerated servers:* The most recent FPGAs that have been developed in the last years incorporate not only programmable logic but also energy-efficient embedded processors (i.e. ARM). Next generation FPGAs will incorporate four or more of high performance energy-efficient processors. VINEYARD will develop novel server blades that will be based on high performance and energy-efficient FPGAs that incorporate multiple ARM cores.
– **Objective 3**: *Development of an integrated programming framework:* This framrwork will be used for the programming of heterogeneous systems consisting of general-purpose processors (CPUs), and accelerators (programmable dataflow engines and FPGAs) based on traditional data-centre programming frameworks (e.g. Spark, Storm, and MapReduce). The framework will hide the complexities of controlling the hardware dataflow accelerators from the user while it will also allow the instantiation of IP modules as pluggable components in the same way that software packages are currently used.

- **Objective 4**: *Development of a run-time scheduler/orchestrator:* This scheduler will control the utilization of the accelerators based on the application requirements (execution time, power consumption, available resources, etc.). It will allow the optimum utilization of the available hardware accelerators based on the use-case constraints. The run-time system that will be developed will be integrated to the run-time systems of the data-centre programming frameworks (MapReduce etc.)
- **Objective 5**: *Development of a novel Virtual-Machine (VM) appliance model for provisioning of data to shared accelerators:* Targeting cloud deployments, the VINEYARD VM appliance will bring both tangible and novel results. The enhanced VINEYARD middleware will augment the functionality of the orchestrator by enabling more informed allocation of tasks to accelerators. The VINEYARD framework will allow the virtualized utilization of hardware accelerators in the servers (Server Function Virtualization - SFV) in the same way that Network Function Virtualization is used to virtualize network functions providing higher flexibility, lower cost and optimized resource utilization.
- **Objective 6**: *Ecosystem Establishment and Support:* Effort will be spent on the establishment of an ecosystem for empowering open innovation based on hardware accelerators as data-centre plugins, thereby facilitating innovative enterprises (large industries, SMEs, and creative start-ups) to develop novel solutions using VINEYARDSs leading edge developments. The ecosystem will bring together existing communities from all relevant stakeholders including providers of hardware-IP technologies, data-centre developers, data-centre operators and more. Stakeholder involvement will be realized through the consortium partners, the representation of communities within the consortium, as well as through the involvement of third-parties based on open calls. This ecosystem will allow the promotion of open, pluggable, custom hardware accelerators that can be used in data centres in the same way that software libraries are currently being utilized. Furthermore, the development of this new ecosystem will enable users from different sectors (open-source communities, universities, research centres, start-ups, etc.) to contribute application-specific accelerators in a repository that can be accessed by data-centre operators.

3 The VINEYARD approach

In this section we will present in more detail the three main building blocks of the project: the programmable dataflow accelerators, the VINEYARD programming framework and the VINEYARD middleware.

3.1 Accelerator-based servers

In the last few years, data centres have experienced a significant increase in the network traffic they handle largely due to the wide adoption of many web

applications such as cloud computing and big data. To cope with this rise in computational and communication demands, data centers have boosted the performance of their server processors which has led to a drastic increase in the power-consumption profiles. Currently, one of the main challenges for data-centre operators is reducing the power costs of their servers that account for over 45% of the overall data-centre power consumption.

Modern server processors contain many levels of caching, forwarding and prediction logic to improve the efficiency of the traditional processor architecture; however the model is inherently sequential with performance limited by the speed at which data can move around this loop.

A dataflow computing model explicitly addresses this issue by minimising and optimising the flow of data. Current dataflow-computing solutions utilise FPGA chip technology, which despite inherent inefficiencies has been shown to lead to orders-of-magnitude lower power consumption and lower data-centre space needs. For example, recent journal publications [5,6] report on production-level use of dataflow computing. The delivery of a Maxeler dataflow machine to JP Morgan, as part of an award-winning initiative described in the Wall Street Journal [7], yielded the computational power of 128 Teraflops (equivalent to over 12,000 high-end x86 control flow cores) within the space and power envelope of a single 40U rack.

Current dataflow engines implemented with FPGAs offer considerable advantages in performance and "performance per Watt". However, FPGAs are a general base technology which has significant limitations and is expensive in silicon area. For example, an FPGA is 18-35x less area efficient than an ASIC at implementing circuits, with a 3-4x higher critical path delay (i.e. decrease in clock frequency). Despite this cumulative 54-140x technology disadvantage, dataflow engines incorporating FPGAs have demonstrated high performance and energy efficiency for a broad range of applications due to the efficiency of the dataflow architecture.

Given such encouraging results, we propose in VINEYARD the development of custom dataflow servers optimised for high-performance, power-efficient implementations of data-centre applications. These servers will maintain the capabilities of FPGAs to implement the dataflow computing paradigm while tackling the sources of inefficiency.

3.2 Programming framework

The current state of the art in programming frameworks lacks a clean solution for integrating the FPGA hardware-software stack with the programming-language runtime system on the host servers. The gap exists from both a semantical and a resource-management perspective. Questions on how accelerator code and state is managed by a high-level functional programming model and runtime system remain largely open. A task abstraction, representing the accelerator as a versioned function to the programming model appears to be the most promising approach [8] but lacks transparency and breaks key desirable properties of functional parallel programming. Furthermore, scheduling, communication and syn-

chronisation in the runtime system are fundamentally influenced by the presence of accelerators. The integration of accelerators with data-management technologies can be more natural due to the declarative nature of queries that can better exploit the data flow model to be implemented in the accelerators.

While bare-metal implementations of MapReduce, OpenCL and other high-level languages for FPGA accelerators have existed for some time [9–11], these implementations are localised and designed to support efficient translation to FPGA hardware rather than integration with the host software stack. The programmability of hardware accelerators (i.e. based on FPGAs) must improve if they are to be part of mainstream computing and data centres.

Combining a host-side programming model with the accelerator programming model in a hybrid solution is a challenging and rather inflexible proposition, both due to semantical conflicts (e.g. differences in memory models) and due to performance implications, notably contention between runtime systems for shared resources [12].

Furthermore, despite efforts to virtualize programmable and hardware accelerators, such as GPUs and FPGAs [13,14], the virtualization methods deployed, notably pass-through and device-drive level, introduce non-trivial performance interference within and between VMs. These are hardly traceable, let alone resolvable by programming models and runtime systems. VINEYARD aspires to address the open challenges in integrating programmable and hardware accelerators to the predominant software stacks used for data analytics in the Cloud:

a) hide the accelerator from the programmer by presenting it as a pure library function, embeddable in query processing, data processing or aggregation tasks, and by extension to analytical libraries written on top of high-level programming models;
b) extend the runtime systems of high-level analytics languages to handle efficiently scheduling, communication, and synchronisation with programmable accelerators; and
c) improve the performance robustness of analytics written in high-level languages against artefacts of virtualization, notably performance interference due to contention on shared resources and hidden noise in hypervisors and hosting VMs.

3.3 Virtualization and Middleware

In principle, the deployment model of accelerators in the data centre can take two basic but different forms: (a) Accelerators can be attached directly to servers and used by local workloads, or (b) accelerators can be shared over the network among many servers and their workloads.

The accelerators – whether GPUs, FPGAs or multicore CPUs – are assigned to tasks which they can perform more efficiently than general-purpose servers. The expected returns in cost, power and execution time are promising, but data movement is a large challenge that underlies the whole proposition. Accelerators

take data from the "slow" CPU path, process them in their customized hardware engines, and return the results either for storage in a file or directly to the memory system or for further processing by other accelerators or servers. The dominant programming frameworks in scale-out data centres have been streamlined to minimize the movement of data; thus, at the end of the day, the value of accelerator-based data centres will be weighed against the cost of the extra data copies that they introduce. With VINEYARD, we will speedup data communication through a system fabric that provides efficient communication primitives, to unify the accelerators with the servers and to reconcile them with the current computing frameworks.

Virtualization support is an additional, significant dimension of data-centre infrastructures. Virtual Machines (and other similar types of technologies such as Containers) offer a mechanism for increasing consolidation of workloads on physical servers and achieving better utilization, isolating software versions and domains, and decoupling administrative domains i.e., clients from providers. Therefore, when examining the potential of accelerators in data centres, it is essential to deal with the implications of, and to accrue the benefits from, Virtual Machines. We note that the presence of tenant VMs inside the data centres is orthogonal to accelerator virtualization [15]. VMs typically access the available hardware resources through a hypervisor, complicating the software segments of I/O stacks and increasing overheads. In addition, sharing the I/O paths among multiple VMs endangers isolation and quality-of-service. Clearly, sluggish and unreliable communication between VMs and accelerators impedes their co-existence in cloud data centres.

Overall, in VINEYARD we will introduce a novel VM appliance model for provisioning of data to shared accelerators. Targeting cloud deployments, this VINEYARD effort can bring both tangible and novel results. The enhanced VINEYARD middleware augments the functionality of the orchestrator, by enabling more informed allocation of tasks to accelerators.

4 VINEYARD Use Cases

Within VINEYARD, an integrated data centre will be developed and will be evaluated through three real-life workloads and industrial benchmarks for financial applications, data management, and scientific computing. The first workload that will be evaluated will be in the domain of *financial applications*. For this reason the Greek Stock Exchange Market will be used as an end-user demanding a) real-time analytics which are necessary for market surveillance and decision management, and b) rapid computations for risk management, as an additional computation step within the trade process chain.

The second workload that will be evaluated will be in the domain of scientific computing, and more specifically in the domain of *computational neuroscience* which aims at better understanding the working of the human brain through simulating biologically plausible neural models. The particular application is a high-performance, high-accuracy simulation of the Olivocerebellar system of the

brain, crucial to the understanding of cerebellar functionality [16]. The Olivocerebellar system is critical for facilitating motor function – among other functionality – in humans. Better modeling and understanding of its function can lead to major breakthroughs in the treatment of various cerebellum-related degenerative diseases such as autism, fragile-X syndrome etc. It will also lead to a deeper understanding of motor control resulting in new automation and robotic technologies, and improved brain-computer interfaces (BCI).

The third workload is a data-management case based on TPC-C (on-line transaction processing (OLTP) benchmark) and TPC-H (decision support benchmark). TPC-C is representative of the transactional workloads run at operational databases of enterprises. It will be run on top of the LeanXcale OLTP database to represent the full stack of enterprise OLTP applications. TPC-H is representative of the analytical workloads run at data warehouses of enterprises. It will be run on top of the LeanXcale OLAP database to evaluate the efficiency improvements for analytical queries. Finally, Linear Road will also be used as a representative workload in IoT applications and will be run on top of the LeanXcale CEP engine to evaluate CEP workloads.

5 Conclusions

The main goal of the VINEYARD project is to develop a new framework for the efficient integration of accelerators into commercial data centres. The VINEYARD project will not only develop novel accelerator-based servers but will also develop all the required systems (hypervisor, middleware, APIs and libraries) that will allow the users to seamlessly utilize the accelerators as an additional cloud resource. The efficient utilization of accelerators in data centres will significantly improve the overall performance of cloud-based applications and will also reduce the energy consumption in the data centres. Finally, VINEYARD aspires to foster the innovation of soft-IP accelerators in the domain of cloud computing by the promotion of a central repository for the hosting of the relevant accelerators.

Acknowledgment

This project has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No 687628.

References

1. In *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update 2014‚2019 White Paper*.
2. Hadi Esmaeilzadeh, Emily Blem, Renée St. Amant, Karthikeyan Sankaralingam, and Doug Burger. Power Challenges May End the Multicore Era. *Commun. ACM*, 56(2):93–102, February 2013.

3. Hadi Esmaeilzadeh, Emily Blem, Renee St. Amant, Karthikeyan Sankaralingam, and Doug Burger. Dark Silicon and the End of Multicore Scaling. In *Proceedings of the 38th Annual International Symposium on Computer Architecture*, ISCA '11, pages 365–376, New York, NY, USA, 2011. ACM.
4. Nikos Hardavellas, Michael Ferdman, Babak Falsafi, and Anastasia Ailamaki. Toward Dark Silicon in Servers. *IEEE Micro*, 31(4):6–15, July 2011.
5. Olav Lindtjorn, Robert Clapp, Oliver Pell, Haohuan Fu, Michael Flynn, and Oskar Mencer. Beyond Traditional Microprocessors for Geoscience High-Performance Computing Applications. *IEEE Micro*, 31(2):41–49, 2011.
6. Stephen Weston, James Spooner, Sebastien Racaniere, and Oskar Mencer. Rapid Computation of Value and Risk for Derivatives Portfolios. *Concurr. Comput. : Pract. Exper.*, 24(8):880–894, June 2012.
7. In *Clark, D. Maxeler makes waves with dataflow design. Wall Street Journal Blog. 13 December 2011.*
8. Javier Bueno, Xavier Martorell, Rosa M. Badia, Eduard Ayguadé, and Jesús Labarta. Implementing OmpSs Support for Regions of Data in Architectures with Multiple Address Spaces. In *Proceedings of the 27th International ACM Conference on International Conference on Supercomputing*, ICS '13, pages 359–368, New York, NY, USA, 2013. ACM.
9. Yi Shan, Bo Wang, Jing Yan, Yu Wang, Ningyi Xu, and Huazhong Yang. FPMR: MapReduce Framework on FPGA. In *Proceedings of the 18th Annual ACM/SIGDA International Symposium on Field Programmable Gate Arrays*, FPGA '10, pages 93–102, New York, NY, USA, 2010. ACM.
10. Peter Athanas, Krzysztof Kepa, and Kavya Shagrithaya. Enabling Development of OpenCL Applications on FPGA Platforms. In *Proceedings of the 2013 IEEE 24th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*, ASAP '13, pages 26–30, Washington, DC, USA, 2013. IEEE Computer Society.
11. Muhsen Owaida, Nikolaos Bellas, Konstantis Daloukas, and Christos D. Antonopoulos. Synthesis of Platform Architectures from OpenCL Programs. In *Proceedings of the 2011 IEEE 19th Annual International Symposium on Field-Programmable Custom Computing Machines*, FCCM '11, pages 186–193, Washington, DC, USA, 2011. IEEE Computer Society.
12. Heidi Pan, Benjamin Hindman, and Krste Asanović. Composing Parallel Software Efficiently with Lithe. In *Proceedings of the 31st ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI '10, pages 376–387, New York, NY, USA, 2010. ACM.
13. Michela Becchi, Kittisak Sajjapongse, Ian Graves, Adam Procter, Vignesh Ravi, and Srimat Chakradhar. A Virtual Memory Based Runtime to Support Multi-tenancy in Clusters with GPUs. In *Proceedings of the 21st International Symposium on High-Performance Parallel and Distributed Computing*, HPDC '12, pages 97–108, New York, NY, USA, 2012. ACM.
14. Wei Wang, Miodrag Bolic, and Jonathan Parri. pvFPGA: Accessing an FPGA-based Hardware Accelerator in a Paravirtualized Environment. In *Proceedings of the Ninth IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis*, CODES+ISSS '13, pages 10:1–10:9, Piscataway, NJ, USA, 2013. IEEE Press.
15. Fei Chen, Yi Shan, Yu Zhang, Yu Wang, Hubertus Franke, Xiaotao Chang, and Kun Wang. Enabling FPGAs in the Cloud. In *Proceedings of the 11th ACM Conference on Computing Frontiers*, CF '14, pages 3:1–3:10, New York, NY, USA, 2014. ACM.

16. Georgios Smaragdos, Sebastian Isaza, Martijn F. van Eijk, Ioannis Sourdis, and Christos Strydis. FPGA-based Biophysically-meaningful Modeling of Olivocerebellar Neurons. In *Proceedings of the 2014 ACM/SIGDA International Symposium on Field-programmable Gate Arrays*, FPGA '14, pages 89–98, New York, NY, USA, 2014. ACM.